Amendments to the Specification:

Please replace paragraph [055] with the following:

--**[055]** The orientation of the image is determined by analyzing the captured image. This process is called disambiguation. One method of disambiguation is described in U.S. Patent Application No. 09/454,526, now US Patent 6,880,755, entitled METHOD AND APPARATUS FOR DISPLAY OF SPATIALLY REGISTERED INFORMATION USING EMBEDDED DATA which discloses substantially the same subject matter as U.S. Patent Application No. 09/455,304, now US Patent 6,678,4[[52]]<u>25</u> entitled METHOD AND APPARATUS FOR DECODING ANGULAR ORIENTATION OF LATTICE CODES, both filed December 6, 1999, which are hereby incorporated by reference.—

Please insert the following NEW paragraphs after paragraph [055]:

[055.1] A disambiguation processes consistent with the present invention will now be described in greater detail using teachings from US Patent 6,678,425 that was incorporated by reference.

[055.2] FIG. 17 of '425 is a flowchart teaching a method to create a composite lattice image pattern for use in determining a quadrant offset angle. The method first selects a seed pixel from the captured image and finds a local minimum in the vicinity of the seed pixel indicating the presence of a glyph. Next the method finds the centroid of this glyph. The method then selects the next seed pixel for analysis at a particular x and y interval from the previously analyzed seed pixel. The particular x and y interval is based on the height and width of the composite lattice image pattern. Next, using the glyph centroid as the origin, the method adds a subsample of the captured image to the composite lattice image pattern. From the resulting composite lattice image pattern the method determines the quadrant offset angle.

[055.3] FIG. 18 of '425 is a flowchart the illustrates a method used to determine a quadrant offset angle using a composite lattice image pattern generated in accordance with the flowchart of FIG. 17 of '425. The method first finds the darkest pixel along an arc between zero and 90 degrees at a distance from the origin equal to the glyph pitch, the distance between adjacent glyphs on the lattice of glyphs, and then finds the centroid of the shape containing this pixel. Once the centroid is found, the method estimates the approximate location of the next minimum along the lattice axis based on the centroid position and the glyph pitch based on the assumption that the lattice axis passes through the centroid and the origin. Using this estimate, the method finds the local minimum around the estimated location, and finds the centroid of the shape containing that minimum. If the last possible minimum has been found, the method fits a straight line, referred to as the axis line, from the origin through the centroids and determines the angle of the axis line, between 0° and 90° and this angle is then offset to fall between -45 degrees and +45 degrees by subtracting 45°.

[055.4] Fig. 23 and Fig. 24 of '425 form a flow chart showing exemplary disambiguation and address decoding processes performed by a computer on the captured image area. The disambiguation process starts by image processing the captured portion of the address carpet to determine the glyph lattice. The glyphs are then decoded as 1's or 0's, which are filled into a binary data matrix having rows and columns corresponding to the glyph lattice rows. The orientation may still be ambiguous with respect to 90° and 180° rotations.

[055.5] Fig. 25 of '425 illustrates a binary data matrix (BDM) 2310 formed from a glyph lattice. Locations in the BDM correspond to locations in the glyph lattice. Each location of the glyph lattice is analyzed to determine which value should be placed in the corresponding location of the BDM. Initially, the BDM is filled with a value, for example φ, which indicates that no attempt has been made to read the glyph. Once the glyph corresponding to a

particular location has been analyzed, φ is replaced by a value indicating the result of the glyph analysis.

[055.6] In Fig. 25 of '425, a B indicates a border location, an X indicates that no interpretable glyph was found at the corresponding location of the glyph lattice, an E indicates a glyph at the edge of the captured image portion, a 0 indicates a back slash glyph, a 1 indicates a forward slash glyph, and d indicates a label code. The area of the matrix corresponding to the captured image is filled with 0's and 1's, the edge is bounded by E's, and the X's correspond to locations that have no readable glyphs.

[055.7] The image capture device might be oriented relative to the substrate at any angle. Therefore, the captured image could be oriented at any angle. Thus, even though a BDM of 0's and 1's is derived from the captured image, it is uncertain whether the BDM is oriented at 0° (i.e., correctly oriented), 90°, 180°, or 270° relative to the original code pattern in the glyph address carpet from which the image was captured. The orientation can be uniquely determined directly from the address codes.

[055.8] After the image has been converted to a BDM, it is processed. The original BDM developed from the captured image is referred to as BDM1. BDM1 is copied and the copy rotated clockwise 90° to form a second binary data matrix, BDM2. By rotating BDM1 by 90°, the rows of BDM1 become the columns of BDM2, and the columns of BDM1 become the rows of BDM2. Additionally, all bit values in BDM2 are flipped from 0 to 1, and 1 to 0.

[055.9] A correlation is separately performed on the odd and even rows of BDM1 to determine whether code in the rows are staggered forward or backward. The correlation is also performed for the odd and even rows of BDM2. The correlation is performed over all the rows of each BDM, and results in correlation value C1 for BDM1 and correlation value C2 for BDM2.

[055.10] Fig. 26 of '425 is a flowchart showing an embodiment of correlation steps 2216 and 2218 of Fig. 24 of '425. The process determines a correlation value

for every other line of a BDM along diagonals in each direction, and sums the row correlation values to form a final correlation value for the odd or even rows. The process is performed on the odd rows of BDM1 to form correlation value C1ODD for BDM1, the even rows of BDM1 to form correlation value C1EVEN for BDM1, the odd rows of BDM2 to form correlation value C2ODD for BDM2, the even rows of BDM2 to form correlation value C2EVEN for BDM2. The BDM that is oriented at 0° or 180° will have a larger CODD + CEVEN than the other BDM. After the process has correlated each adjacent row, the correlation value C_RIGHT indicates the strength of the correlation along the diagonals to the right. Similar processing is performed on diagonals running from the upper right to lower left to develop correlation value C_LEFT. After correlating the right and left diagonals to determine C_RIGHT and C_LEFT, a final correlation value C is determined by subtracting C_LEFT from C_RIGHT. For example, if odd rows for BDM1 are processed, the C value becomes C1ODD for BDM1. In addition, correlations are performed for the odd and even rows of BDM1 and the odd and even rows of BDM2. From this information, the correlation value C1 for BDM1 is set to C1EVEN + C1ODD, and the correlation value C2 for BDM2 is set to C2EVEN + C2ODD.

[055.11] For each BDM, four correlation values are developed: 1) odd rows, right to left, 2) odd rows, left to right, 3) even rows, right to left and 4) even rows, left to right. From these correlation values, the strongest correlation value for the even rows, and strongest correlation value for the odd rows is chosen, and these become CEVEN and CODD for that BDM (steps 2216 of '425and 2218 of '425). CEVEN and CODD are then added to form a final C correlation value for that BDM. The BDM with the strongest correlation value is the BDM that is oriented at either 0° or 180° because of the relative orientation of the codes in the odd and even rows. Thus, two aspects of the chosen BDM are now established: which direction every other line of codes is staggered, and that the BDM is oriented horizontally, at either 0° or 180°. Another correlation process, at step 2230 of '425 is performed to determine which

direction the code in each line runs (as opposed to which way the code is staggered).

[055.12] The codes in the odd lines are staggered in one direction, and the codes in the even lines are staggered in the other. This staggering property of the code, in conjunction with knowing the respective codes that run in the odd lines and even lines, allows determination of the proper 0° orientation of the BDM.

[055.13] Note that if C1 is greater than C2, then BDM1 is selected for further processing. C1 being greater than C2 indicates that the one-dimensional codes of BDM1 are most strongly correlated and are, therefore, oriented at either 0° or 180°. If C2 is greater than C1, then BDM2 is selected for further processing, because the higher correlation indicates that BDM2 is oriented at either 0° or 180°. Thus, the correct BDM has been found. However, it still must be determined whether the selected BDM is at 0° (i.e., oriented correctly), or rotated by 180°.

[055.14] Fig. 24 of '425 is a flowchart showing the steps to determine the address of the captured area of the glyph carpet. Preferably, bit positions along a diagonal in the BDM, when the BDM is oriented at 0°, have the same value at every other row. This results in a first code sequence for the odd rows and a second code sequence for the even rows.

[055.15] Expected codes (pseudo noise) for rows staggered forward and for rows staggered backward are cross correlated with the BDM to establish the best match of the glyph sequence with pseudo noise sequence for the odd and even rows. The four correlations develop four pairs of peak correlation and position values that disambiguates the rotation of the BDM.